EAGLE

and the

High Level Architecture

Sponsors:

OSD PA&E

U.S. Army TRADOC Analysis Center

U.S. Army National Simulation Center

Jack Ogren

The MITRE Corporation

jogren@mitre.org

April 1997

MITRE

Contexts
Eagle & the High Level Architecture

- Eagle Design Summary
- Summary of Effort & Design
- Technical Approach
- Class Structures
- Interactions
- Event Synchronization
- Summary

Contexts

Eagle & the High Level Architecture

- ❹ **Eagle Design Summary**
- 0 **Summary of Effort & Design**
- 0 **Technical Approach**
- 0 **Class Structures**
- 0 **Interactions**
- 0 **Event Synchronization**
- 0 **Summary**

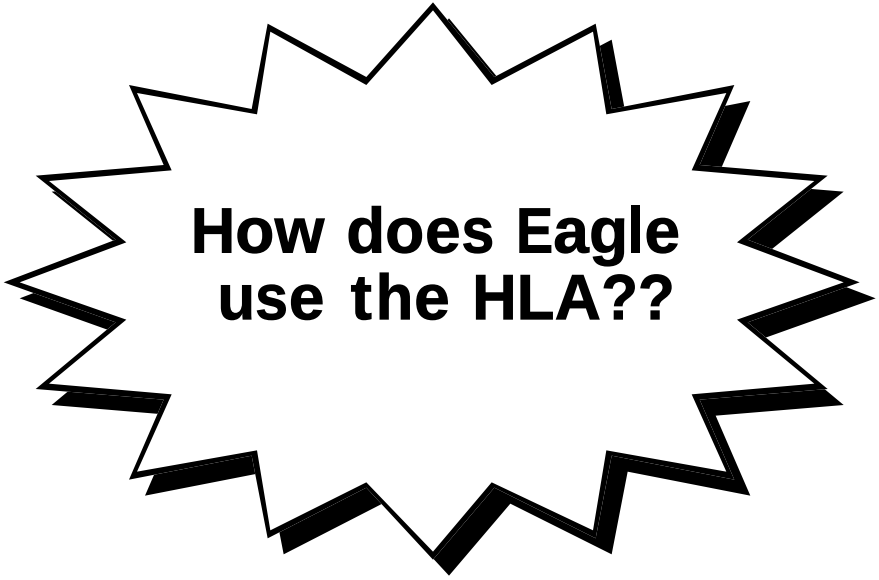
Eagle Design -- Summary

- Used by TRAC as a combat development analysis tool to study corps and division level force effectiveness issues. (Not a Training Simulation)
- Characteristics
  - Corps & below level simulation
  - Resolution to Battalion or Company (Entity level w/BDSD)
  - Deterministic (Stochastic w/BDSD)
  - Hybrid event Structure
    - = Attrition/Ground Movement/Detection/C2 are Time-stepped
    - = Air/ADA interactions are Event-driven
  - Integrates Artificial Intelligence methods and conventional combat modeling algorithms
    - Object-Oriented
    - Embedded AI Systems (expert systems)
    - Symbolic decision making
    - Lisp/KEE Programming environment
  - Command and Control modeled explicitly.
    - Units execute orders and pass information based on a Battlefield Management language.

Contexts

Eagle & the High Level Architecture

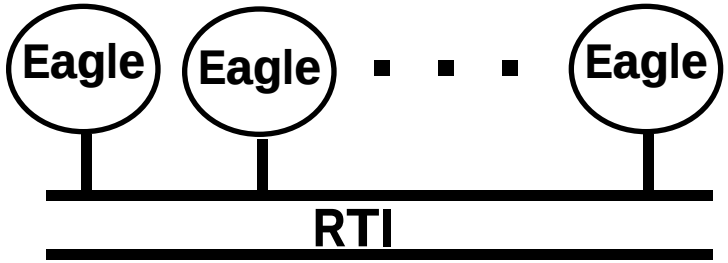
- Eagle Design Summary
- ❹ Summary of Effort & Design
- Technical Approach
- Class Structures
- Interactions
- Event Synchronization
- Summary



Eagle & the HLA
Summary of Effort - 3 Separate Federations

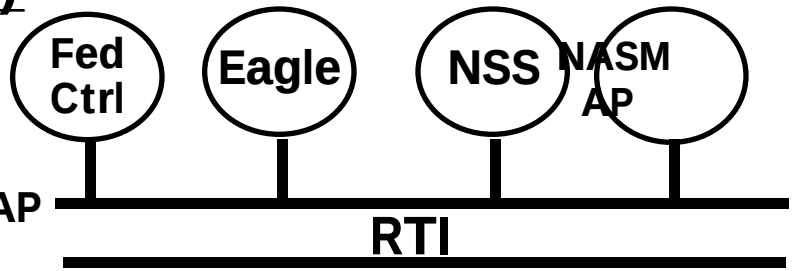
Eagle Early Analysis Experiment

Distributed Units -
Eagle Combat units interact using the RTI.



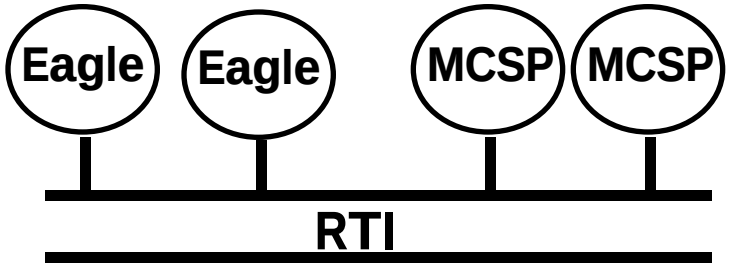
Joint Training Federation (JTFp)

Distributed Functionality-
Army Combat units - Eagle
Navy Combat units - NSS
Air Force Combat units - NASM AP
Federation Controller



Eagle to MCSP/Beta - WarLab

Eagle Combat Hqs divest Cognitive Processing to Live Players using MCSP



Design facilitate running as standalone, individual or combined federations

Distributed Eagle Goal & Basic Design

- **GOAL: Decrease computation load on single processor while minimizing interactions and maintaining the same temporal, tactical, organizational and spacial consistency that currently exists on a single processor.**
  - **Maintain current validated models.**
- **DESIGN: Combat units are distributed among multiple Eagle simulations on the network.**
  - **Each simulation maintains its own set of core services**

    terrain, terrain evaluation, attrition, tactics DB ... etc
  - **Each simulation reflects all units not owned by the simulation.**
  - **The distributed design maintains consistency between:**

    Services on simulations (Terrain conditions).

    Combat units and their reflected representations.

Functionally Distributed Eagle Goal & Basic Design

- **GOAL: Allow externally generated Objects (combat or environmental) to interact with Eagle generated combat units while maintaining the same or better temporal, tactical, organizational and spacial consistency that currently exists on a single processor.**
- **DESIGN:**
  - **Eagle provides all ground combat functionality for the federation. All ground combat units are reflected.**
  - **Functionality within Eagle is replaced by functionality provided by the confederation members.**
    - **Eagle replaces normal fixed wing operations by**
      - **Subscribing to the federation air objects (classes)**
      - **Publishing & subscribing to interactions between the ground & air players (air to ground / ground to air)**

Eagle to MCSP/Beta - Goal & Basic Design

- **GOAL: Allow Live Players, using the MCSP/Beta, to replace the normal command and control activities of a simulated command post.**
- **DESIGN:**
  - **All physical activities of the command post are played in the simulation.**
  - **Cognitive activities of the command post are divested to the live players. Two way interactions are allowed**
  - **All information presented to the live players and their decisions are saved for analysis.**
  - **The simulated command post maintains a memory of its current state and decisions and can take over from the live players at any time.**

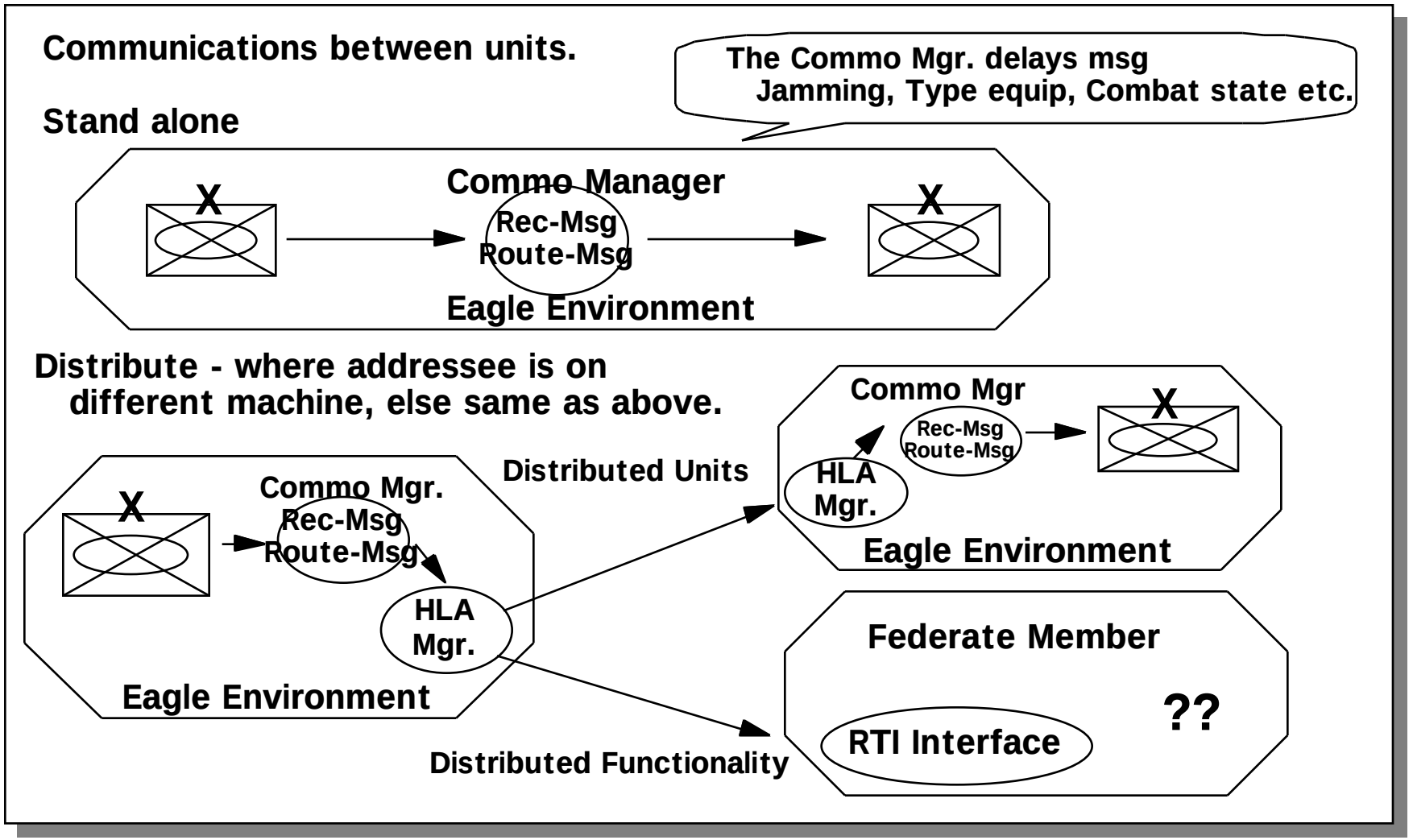
Contexts

Eagle & the High Level Architecture

- Eagle Design Summary
- Summary of Effort & Design
- ❹ Technical Approach
- Class Structures
- Interactions
- Event Synchronization
- Summary

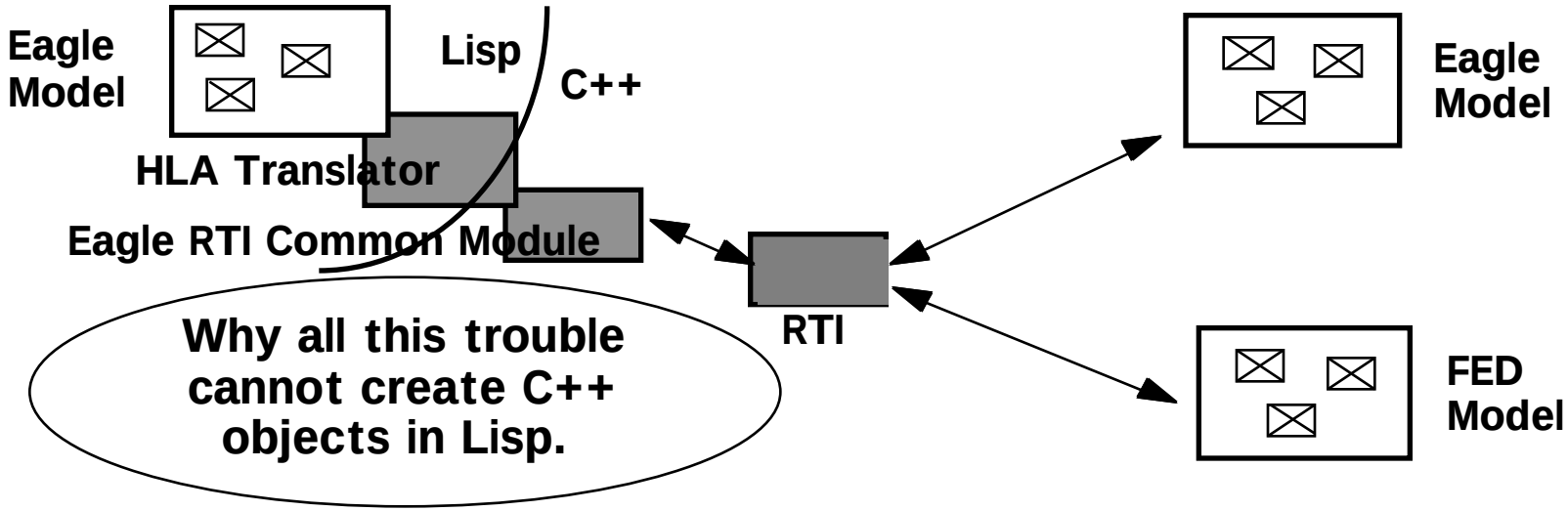
{ Within Eagle and Between Eagle & the RTI

Technical Approach within Eagle - Example



Technical Approach between Eagle & the RTI

- A new Eagle Service (HLA Translator) is provided within the Eagle framework to provide the interface between Eagle objects & the RTI.
  - Very little was modified within the actual Eagle code. Most of the work was focused on the translator (11,000 lines of code).
- Each Eagle simulation HLA Translator Service (lisp) communicates with the RTI through an Eagle RTI Common module (C++). (app 4,000 lines of code)
- The Eagle RTI Common module provides the interface with the RTI -(RTI ambassador & Federate Ambassador).

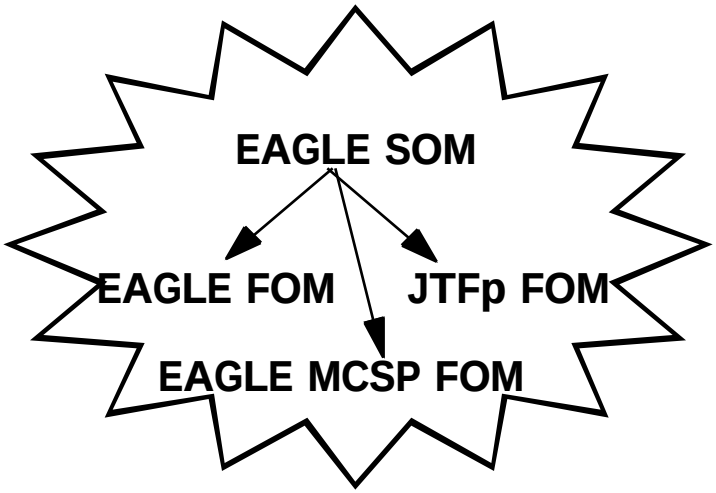


Contexts

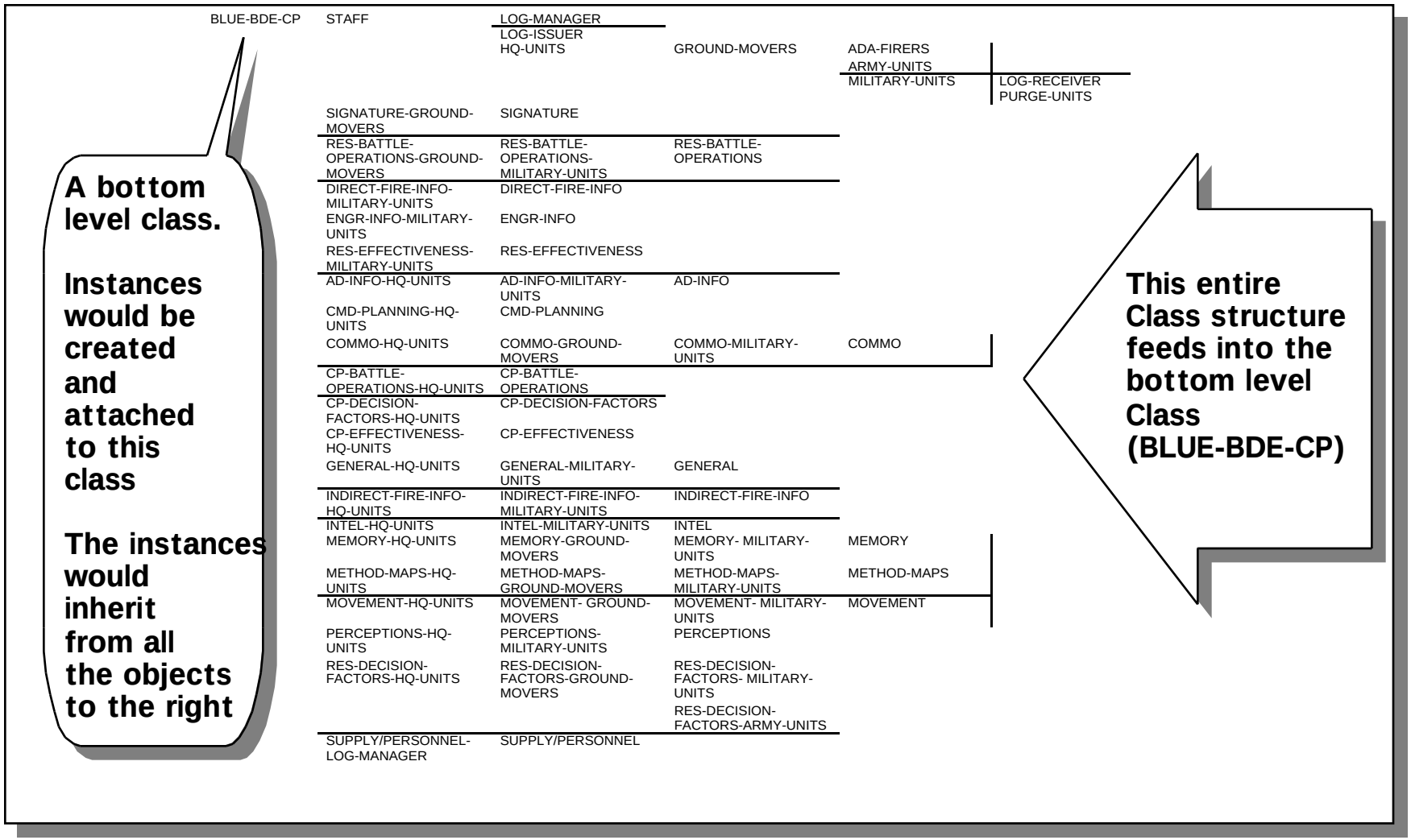
Eagle & the High Level Architecture

- Eagle Design Summary
- Summary of Effort & Design
- Technical Approach
- ❹ Class Structures
- Interactions
- Event Synchronization
- Summary

Software Object Model and Federation Object Model



Eagle Class Structure
Normal structure in model (one example)



Eagle Class Structure ❹ Eagle SOM ❹
Distributed Eagle FOM && JTFp FOM

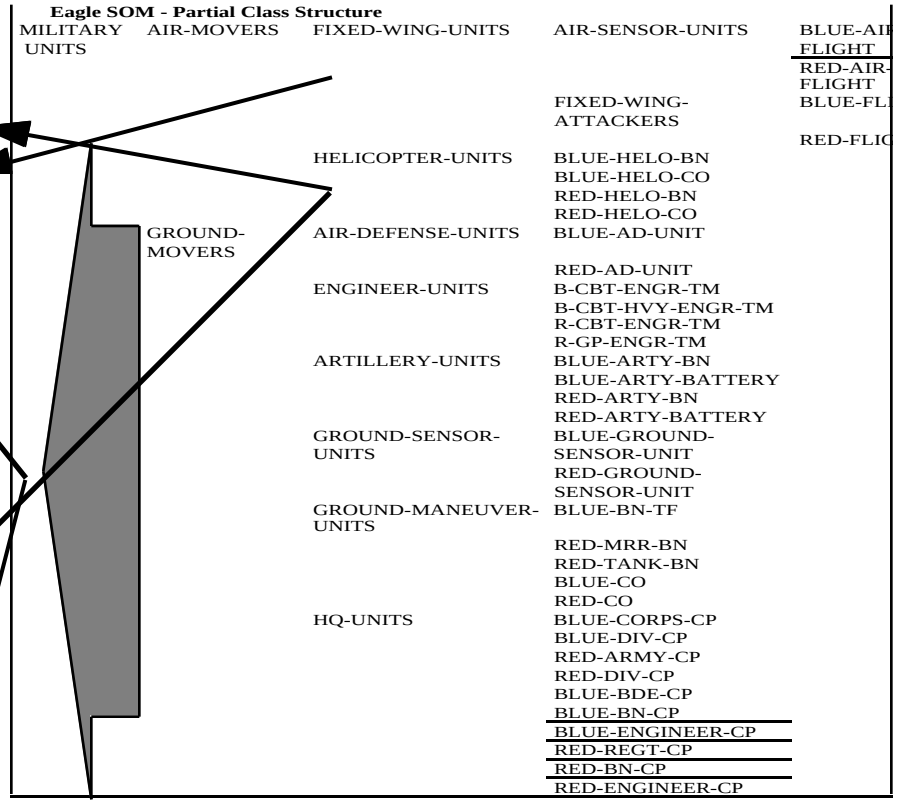
Distributed Eagle FOM
(partial)

Base Class	1st SubClass	2nd SubClass
Military Units	Ground Movers	
	Air Movers	Air Maneuver
		Fixed Wing

JTFp FOM
(partial)

Base Class	1st SubClass	2nd SubClass
Player	Air Player	Ballistic Missile
		Aircraft
		Flight
	Ground Player	Fixed Site
		Mobile Ground
		Aggregate Ground
	Float Player	

Eagle SOM Class Structure
(partial)



Class Structure

Attributes used to define Reflected Units

Distributed Eagle FOM

(Partial - Ground Mover)

OBJECT	ATTRIBUTE
MILITARY-UNITS	Battlefield Operating System
	Higher Hdq's name
	Depth of unit (meters)
	Direction of movement of the unit
	Disaggregated boolean
	Echelon
	Percent Effective
	Effectiveness State
	Frontage of Unit (meters)
	Latitude
	Longitude
	Name
	Operational Activity
	Orientation of weapons
	Phantom boolean
	Purged boolean
	Quantity of Sytems on-hand by type
	Route segment
	Side
	Size
	Systems requirements code
	System configuration
	Task
	Unit Type
GROUND-MOVERS	Air Defense Controller
	Air Defense Network boolean
	Air Defense Status
	Assignment
	Command Assignment
	Command Unit boolean
	Indirect Fire Systems on-hand
	Number of Indirect Fire Units
	Prioritized Enemy List
	Signature - counter battery
	Signature - counter mortar
	Signature - combat
	Signature - communications
	Previously detected boolean
	Signature - IR
	Signature - Moving Target Radar
	Signature - photo
	Net speed from last time step
	System weight factor

JTFp FOM

(Partial - Aggregate Ground Player)

OBJECT	ATTRIBUTE
Player	entity name
	federate id
	affliliation
	motion type
	voice nets
	jtids nets
	trap tre
	comander type
Aggregate Entity	radar cross section
	radar detectable
	elint detectable
	comint detectable
	ir detectable
	photoint detectable
	air to air engageable
	surf to air engageable
	air to surf engageable
	surf to surf engageable
	composition
Dead Reckoned Players	time at last cse change
	lat at last cse change
	lng at last cse change
	alt at last cse change
	cse at last cse change
	hspd at last cse change
	vspd at last cse change
Aggregate Ground Player	depth
	front
	orientation

Number of Attributes to define a typical ground combat unit in Eagle: ~ 400

Number of Attributes to reflect units in Distributed Eagle:

	Define	Update
Ground Movers	43	31
Air Maneuver	35	26
Fixed Wing	33	22

Number of Attributes to reflect Aggregate Ground Units in JTFp:

	Define	Update
Ground Movers	29	17

Attributes used for detection's

Contexts

Eagle & the High Level Architecture

- Eagle Design Summary
- Summary of Effort & Design
- Technical Approach
- Class Structures
- ❹ Interactions
- Event Synchronization
- Summary

Distributed Eagle Interaction Structure

23 interaction types - total with subtypes: 65

- **Unit Interactions**
  - **Detection (implicit)**
  - **Direct Fire**
  - **Indirect Fire**
  - **Air to ground**
  - **Ground to air**
  - **Communications**

- **Service Interactions**
  - **Event Queue Management**
    - **Add/remove events**
  - **Control Measures Database consistency**
    - **Create CM**
  - **Terrain Database consistency**
    - **false targets**
    - **register engr work**
  - **Attrition**
    - **Ammo use**
    - **Suppression**

JTFp Interactions that involve Eagle

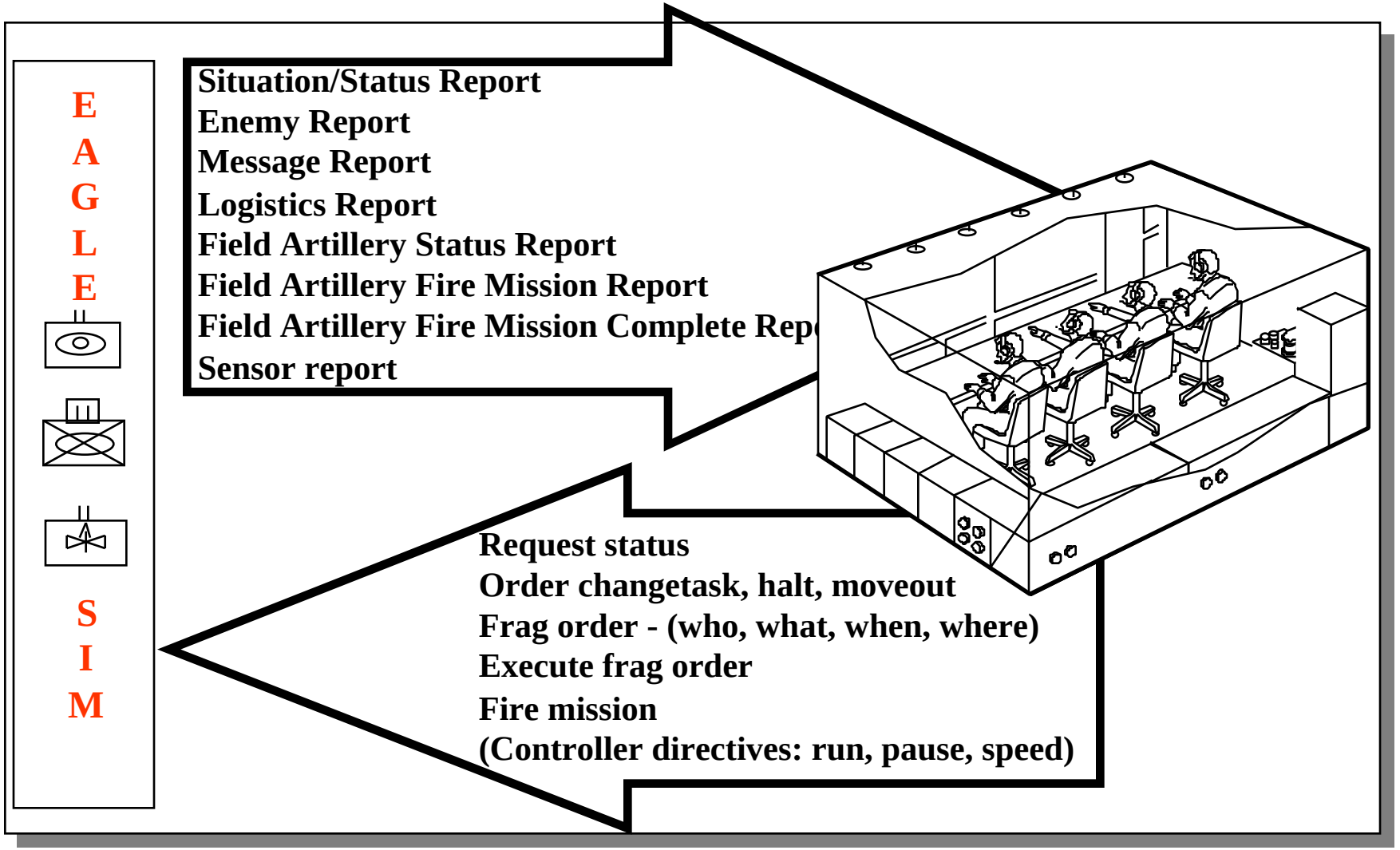
- **Unit Interactions**
  - **Detection (implicit)**
  - **Indirect Fire**
  - **Air to surface**
  - **Surface to air**
  - **Communications**
- **Federation Management Interactions**
  - **Initialization**
- **Environmental**
  - **LOS**

Interaction	Initiating	Receiving
TBM Warming	Player	Eagle Commo Manager - Player
TBMLaunchAlert	Player	Eagle Commo Manager- Player
Situation Report	Player - Eagle Commo Manager	Player (JTF HQ)
RequestAirSupport	Player - Eagle Commo Manager	Player (JTF HQ)
AirToAggregate GroundEngage	Player	Eagle Attrition Manager
DiscreteGroundToAirEngage	Player	Eagle Attrition Manager
AggregateGround ToAirEngage	Eagle Attrition Manager	Player
Get & Return LOS	Eagle Terrain Manager	Surface Cover
Initialize Federation	Federation Status	Eagle Federation Status
Execute Federation	Federation Status	Eagle Federation Status

JTFp has defined 24 Interactions
Eagle subscribes/published to 10

EAGLE To MCSP Interactions

10 interactions defined (9 to MCSP, 1 to Eagle)



Contexts

Eagle & the High Level Architecture

- Eagle Design Summary
- Summary of Effort & Design
- Technical Approach
- Class Structures
- Interactions
- ❹ Event Synchronization
- Summary

The Challenge

To maintain a consistent view of time within Eagle regardless of the types of simulations that are participating in a federation.

Simulations that use RTI Time Mgt
1. Use time steps to advance time timesteps <=> Eagles
2. Use events to advance time (NSS,NASM)

Coordinated

Running as fast as possible or scaled real time

paced - yes/no

Simulations that do not use RTI Time Mgt (DIS)

Independent

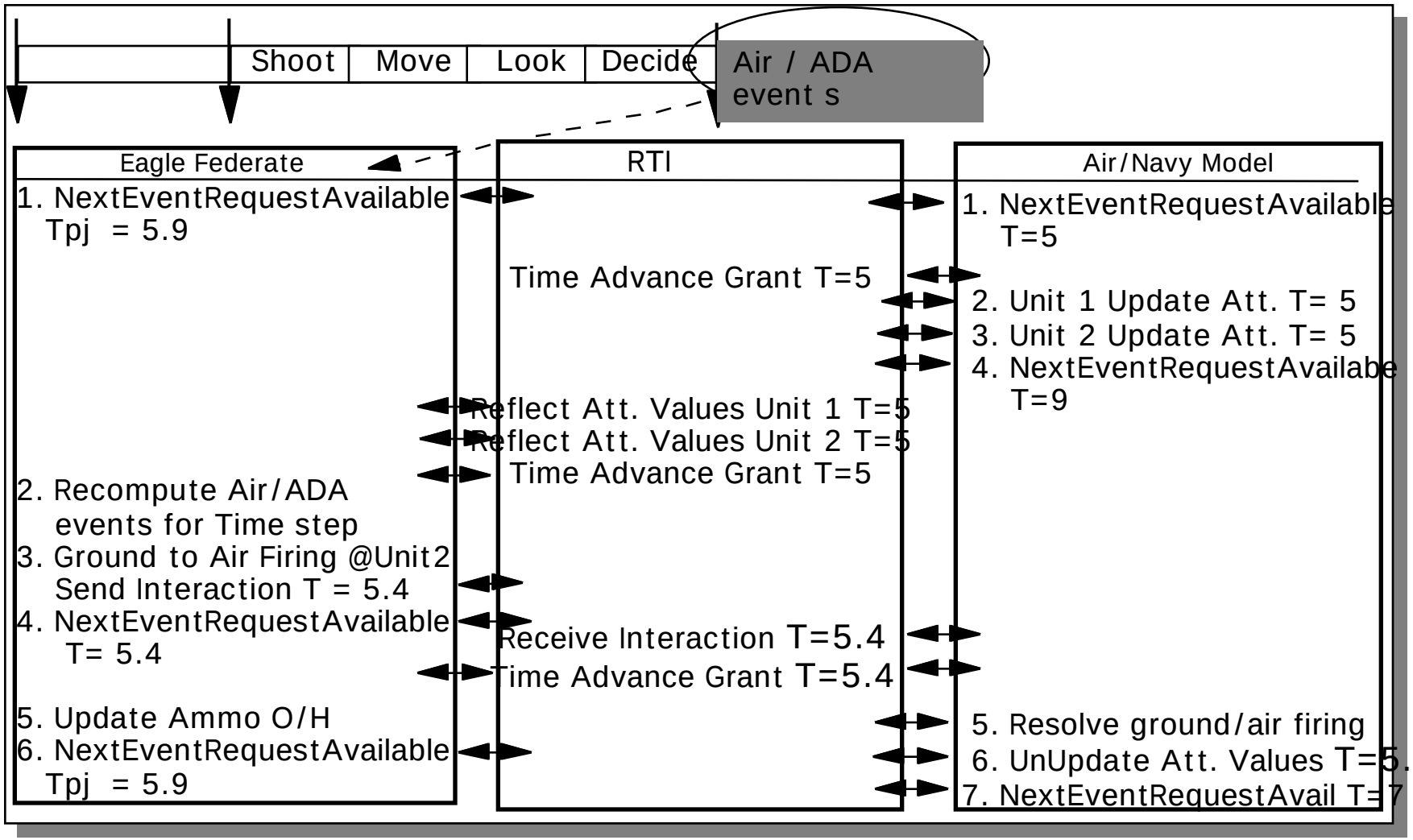
Real time

paced - yes

Eagle has a hybrid event structure that relies on both the notion of continuous time using time steps (1 to 5 mins) and the projecting of discrete events limited to the duration of the time step.

EAGLE TIME MANAGEMENT

Event Update Process -(Zero Lookahead Services)



Event Queue Management

- Subordinate Eagle's simulation event queue to that of the RTI
  - Approval to execute an Eagle event must come from the RTI.
    - =Request may be denied .
- Ability to respond to new external events from RTI.
    - =Eagle simulation event queue repair.
    - =Updating of aircraft situation within Eagle's timestep.
    - =Reallocation of ADA to respond to aircraft projected flight paths.
- Maintaining of Federate Time
- Causality & Consistency was maintained !!!!!

Contexts

Eagle & the High Level Architecture

- Eagle Design Summary
- Summary of Effort & Design
- Technical Approach
- Class Structures
- Interactions
- Event Synchronization
- ❹ Summary

Summary - Technical

Eagle & the High Level Architecture

- **Code Changes (< 3.0%)**
  - **Basic model is app. 750,000 lines of code**
  - **Added Translator Interface app. 16,000 lines of Lisp code**
  - **Added 3,742 lines of C++ interface code.**
- **Time**
  - **Initial design & coding of Distributed Eagle using ALSP = 10 months**
  - **SOM/FOM development = 1 months**
  - **Modify the ALSP interface code = 3 months**
  - **Creation of C++ interface code = 1 month**
  - **Testing = 1 month**
- **Eagle's Design (Architecture) & Object Oriented approach facilitated transition.**

Summary - WHY????

Eagle & the High Level Architecture

- **Move to Distributed Simulation -**
  - **Increase speed & number of units able to play.**
- **Ability to share data with other simulations.**
  - **Able to use Military Service sponsored simulations that meet their standards of validation.**
    - **=We get their validated and verified modeling algorithms and data for the representation of their systems.**
      - **Better representation of Air - completely replaced**
      - **Better physical models - flying of TBM's**
      - **New Functionality - Ships**
- **Through Eagle/MCSP Interface able to improve (?) the simulation by allowing live players to replace the simulated command entities in Eagle.**